

# Rigorous constraint satisfaction for sampled linear systems

Moritz Schulze Darup<sup>†</sup>

**Abstract.** We address a specific but recurring problem related to sampled linear systems. In particular, we provide a numerical method for the rigorous verification of constraint satisfaction for linear continuous-time systems between sampling instances. The proposed algorithm combines elements of classical branch and bound schemes from global optimization with a recently published procedure to bound the exponential of interval matrices.

**Keywords.** Sampled linear system, state and input constraints, branch and bound, interval arithmetic, matrix exponential.

## 1 Introduction and Problem Statement

We consider the continuous-time linear system

$$\dot{x}(t) = A x(t) + B u(t), \quad x(0) = x_0 \quad (1)$$

with state and input constraints of the form

$$x(t) \in \mathcal{X} \quad \text{and} \quad u(t) \in \mathcal{U} \quad \text{for every } t \in \mathbb{R}_0 \quad (2)$$

under piecewise constant control

$$u(t) = u(t_k) \quad \text{for every } t \in [k \Delta t, (k+1) \Delta t), \quad (3)$$

where  $\Delta t > 0$  denotes the sampling time and where  $t_k := k \Delta t$  for every  $k \in \mathbb{N}$ . The sets  $\mathcal{X} \subset \mathbb{R}^n$  and  $\mathcal{U} \subset \mathbb{R}^m$  are assumed to be convex and compact polytopes containing the origin as an interior point. During controller design (and controller evaluation), system (1) is usually replaced by the discrete-time system

$$x(t_{k+1}) = \hat{A} x(t_k) + \hat{B} u(t_k), \quad x(0) = x_0 \quad (4)$$

with  $\hat{A} := \exp(A \Delta t)$  and  $\hat{B} := \int_0^{\Delta t} \exp(A \tau) d\tau B$ . While the discretized system and the continuous-time system coincide at all sampling instances, it is well-known that the continuous-time trajectory may violate the state constraints even though the discrete-time counterpart does not (see, e.g., the motivating example in [1]). This potential problem can be prevented by computing adapted constraints for the discretized system

---

<sup>†</sup>M. Schulze Darup is with the Control Group, Department of Engineering Science, University of Oxford, Parks Road, Oxford OX1 3PJ, UK. E-mail: [moritz.schulzedarup@rub.de](mailto:moritz.schulzedarup@rub.de).

such that constraint satisfaction of (4) w.r.t. the adapted constraints implies constraints satisfaction of (1) w.r.t. the original constraints (2) (see, e.g., [1, 2, 3]).

Comparing the methods for the computation of adapted constraints in [1, 2, 3], it is peculiar that the procedures in [2] and [3] both rely on similar non-convex optimization problems (cf. [2, Thm. 5] and [3, Eq. (15)]). Roughly speaking, the underlying problem reads as follows. For a given state  $x_0 \in \mathcal{X}$  and input  $u_0 \in \mathcal{U}$  such that  $\hat{A}x_0 + \hat{B}u_0 \in \mathcal{X}$  (i.e., the discretized systems satisfies the constraints), we are interested in checking whether the trajectory of the continuous-time systems violates the state constraints for some  $t \in (0, \Delta t)$ .

More formally, the problem of interest can be described along the following lines. First note that the polytope  $\mathcal{X}$  can be written in the form

$$\mathcal{X} = \{x \in \mathbb{R}^n \mid Hx \leq \mathbf{1}\},$$

where  $H \in \mathbb{R}^{p \times n}$  and where  $\mathbf{1} \in \mathbb{R}^p$  is a vector with all entries equal to 1. Now, let  $\varphi(t, x_0, u_0)$  denote the solution of (1) at time  $t \in [0, \Delta t]$  for an initial condition  $x_0 \in \mathcal{X}$  and a control action  $u_0 \in \mathcal{U}$ . Then, the trajectory of the continuous-time system does obviously not violate the state constraints for any  $t \in [0, \Delta t]$  if

$$\max_{j \in \mathbb{N}_{[1, p]}} \max_{t \in [0, \Delta t]} e_j^T H \varphi(t, x_0, u_0) \leq 1, \quad (5)$$

where  $e_j \in \mathbb{R}^p$  is the  $j$ -th Euclidean unit vector. Taking into account that  $\varphi(t, x_0, u_0)$  reads

$$\varphi(t, x_0, u_0) = \exp(A t) x_0 + \int_0^t \exp(A \tau) d\tau B u_0 \quad (6)$$

for every  $t \in [0, \Delta t]$ , it is easy to see that  $e_j^T H \varphi(t, x_0, u_0)$  is in general not concave (nor convex) in  $t$ . Hence, verifying whether (5) holds (or not) is a multivariate non-convex optimization problem (OP). Fortunately, the l.h.s. in (5) can be easily decomposed into  $p$  univariate OPs of the form

$$f^* := \max_{t \in [0, \Delta t]} f(t), \quad (7)$$

where  $f : [0, \Delta t] \rightarrow \mathbb{R}$  is given by

$$f(t) := h^T \left( \exp(A t) x_0 + \int_0^t \exp(A \tau) d\tau B u_0 \right) \quad (8)$$

with  $h \in \mathbb{R}^n$ . Clearly, (5) holds if  $f^* \leq 1$  results from (7) for every  $h \in \{H^T e_1, \dots, H^T e_p\}$ .

Following the argumentation in [2] (and [3]), although (7) is non-convex, it can be solved reliable since it is the search of the maximum of a scalar function on a scalar compact domain. While this observation is true, we can provide more elaborated solution strategies for (7) based on the special structure of the objective function in (8). In this paper, we address the rigorous (or global) solution of (7) using interval arithmetic (IA). More precisely, we intend to identify non-decreasing, non-increasing, convex and concave segments of  $f(t)$  on  $[0, \Delta t]$  based on interval inclusions for the first and second

time-derivative of  $f(t)$ . Clearly, for such segments, local maxima can be easily computed and subsequently finding the global maximum is straightforward.

The paper is organized as follows. We state basic notation and preliminaries in Sect. 2. The main result of the paper, i.e., a tailored branch and bound algorithm for the rigorous solution of (7) is presented in Sect. 3. Finally, the proposed method is illustrated with some examples in Sect. 4 before giving conclusions in Sect. 5.

## 2 Notation and Preliminaries

As mentioned in the introduction, we exploit IA to provide interval inclusions for  $f(t)$  and its derivatives

$$\frac{df(t)}{dt} := f'(t) \quad \text{and} \quad \frac{d^2f(t)}{dt^2} := f''(t).$$

IA can be understood as the extension of operations associated with real numbers, like addition or multiplication, to intervals (see, e.g., [4, Sect. 2.2]). In this paper, we only require a few interval operations summarized in the following lemma.

**Lemma 1** ([4, Eqs. (2.14) and (2.19)]). *Let  $[c] = [\underline{c}, \bar{c}] \subset \mathbb{R}$  and  $[d] = [\underline{d}, \bar{d}] \subset \mathbb{R}$  be intervals with  $\underline{c} \leq \bar{c}$  and  $\underline{d} \leq \bar{d}$ . Define the intervals*

$$\begin{aligned} [c] + [d] &:= [\underline{c} + \underline{d}, \bar{c} + \bar{d}] \quad \text{and} \\ [c] \times [d] &:= [\min\{\underline{c}\underline{d}, \underline{c}\bar{d}, \bar{c}\underline{d}, \bar{c}\bar{d}\}, \max\{\underline{c}\underline{d}, \underline{c}\bar{d}, \bar{c}\underline{d}, \bar{c}\bar{d}\}]. \end{aligned}$$

*Then,  $c + d \in [c] + [d]$  and  $cd \in [c] \times [d]$  for every  $c \in [c]$  and every  $d \in [d]$ .*

The rules in Lem. 1 can also be applied to compute the sum (or the multiplication) of an interval  $[c]$  and a real number  $d \in \mathbb{R}$ . In this case,  $[d]$  can be construed as a degenerated interval with  $\underline{d} = \bar{d} = d$ . Moreover, by setting  $[d] = [c]$ , the interval multiplication can be used to evaluate  $[c]$  raised to the power of  $\kappa \in \mathbb{N}$ . However, tighter inclusions result for the calculation rule given in [4, Eq. (3.10)]. In fact, we find  $c^\kappa \in [c]^\kappa$  for every  $c \in [c]$ , where

$$[c]^\kappa := \begin{cases} [\underline{c}^\kappa, \bar{c}^\kappa] & \text{if } \underline{c} > 0 \text{ or } \kappa \text{ is odd,} \\ [\bar{c}^\kappa, \underline{c}^\kappa] & \text{if } \bar{c} < 0 \text{ and } \kappa \text{ is even,} \\ [0, |[c]^\kappa|] & \text{if } 0 \in [c] \text{ and } \kappa \text{ is even,} \end{cases}$$

and where the magnitude of  $[c]$  is defined as  $|[c]| := \max\{|\underline{c}|, |\bar{c}|\}$ . In addition, we define the width of an interval as  $w([c]) := \bar{c} - \underline{c}$ . IA can be easily extended to interval vectors and interval matrices. For two interval matrices  $[C] = [\underline{C}, \bar{C}]$  and  $[D] = [\underline{D}, \bar{D}]$  of appropriate size, the sum  $[C] + [D]$  and the multiplication  $[C][D]$  are understood component-wise. Analogously, the magnitude  $|[C]|$  is defined component-wise, i.e.,  $(|[C]|)_{ij} := |[\underline{C}_{ij}, \bar{C}_{ij}]|$ . Finally, the infinity norm of an interval matrix is defined as the maximum of the norms of the contained real matrices, i.e.,  $\|[C]\|_\infty := \max_{C \in [C]} \|C\|_\infty$ . It is easy to see, that this definition implies  $\|[C]\|_\infty = |||[C]|||_\infty$ . Computing interval inclusions for (8) will mainly build on interval inclusions for matrix exponentials, which can be calculated as follows.

**Theorem 2** ([5, Thm. 4.3]). *Let  $[C] = [\underline{C}, \overline{C}]$  be an interval matrix with  $\underline{C}, \overline{C} \in \mathbb{R}^{q \times q}$ . Let  $k, l \in \mathbb{N}$  be such that  $2^l(k+2) > \|[C]\|_\infty$ . Define  $[C^*] := \frac{1}{2^l}[C]$ ,*

$$[D^*] := I_q + \frac{[C^*]}{1} \left( I_q + \frac{[C^*]}{2} \left( \dots \left( I_q + \frac{[C^*]}{k} \right) \dots \right) \right) \\ + \frac{\|[C^*]\|_\infty^{k+1}}{(k+1)!(1 - \frac{\|[C^*]\|_\infty}{k+2})} [-I_q, I_q],$$

and  $[D] := [D^*]^{2^l}$ . Then  $\exp(C) \in [D]$  for every  $C \in [C]$ .

Note that there exist many ways to evaluate  $[D^*]^{2^l}$  as occurring in Thm. 2. In [5, p. 61], the authors propose to use  $l$  successive interval square operations, i.e.,

$$[D^*]^{2^l} = \left( \dots ([D^*]^2) \dots \right)^2.$$

An efficient procedure for the computation of the square of an interval matrix is presented in [6, Sect. 6].

### 3 Rigorous Solution via Interval Arithmetic

In the following, we present a tailored method for the rigorous solution of the non-convex OP (7). Before describing the algorithm, we have to stress that there exists a number of situations where (7) can be solved analytically. In this context, note that (8) can be rewritten as

$$f(t) = h^T \left( \int_0^t \exp(A\tau) d\tau (Ax_0 + Bu_0) + x_0 \right) \quad (9)$$

using the identity  $\int_0^t \exp(A\tau) d\tau A + I_n = \exp(At)$ . Obviously, trivial solutions result if  $Ax_0 + Bu_0 = 0$ ,  $A = 0$ , or  $h = 0$ . In addition, an analytical solution of (7) is straightforward if  $h$  is an eigenvector of  $A^T$ , i.e., if  $h^T A = \lambda h^T$  for some  $\lambda \in \mathbb{R}$ . To see this, note that the time-derivatives of (9) are given by

$$f'(t) = h^T \exp(At) (Ax_0 + Bu_0) \quad \text{and} \quad (10)$$

$$f''(t) = h^T A \exp(At) (Ax_0 + Bu_0). \quad (11)$$

Thus,  $h$  being an eigenvector implies  $f''(t) = \lambda f'(t)$  for every  $t \in [0, \Delta t]$ , which eventually leads to a monotone function  $f$ . Consequently, we obtain  $f^* = \max\{h^T x_0, f(\Delta t)\}$ . Finally, the solution of (7) may be trivial if  $A$  is nilpotent, i.e., if there exists an  $r \in \mathbb{N}$  (with  $1 \leq r \leq n$ ) such that  $A^k \neq 0$  for  $k \in \{0, \dots, r-1\}$  and  $A^k = 0$  for  $k \geq r$ . In this case,  $f$  can be rewritten as the polynomial

$$f(t) = h^T \left( x_0 + \sum_{k=1}^{r-1} \frac{A^k x_0 + A^{k-1} B u_0}{k!} t^k + \frac{A^{r-1} B u_0}{r!} t^r \right). \quad (12)$$

If an analytical solution is not obvious, a numerical procedure to solve (7) may be required. We propose Alg. 1 further below to compute  $\epsilon$ -optimal solutions to (7) according to Def. 1.

**Definition 1.** Let  $\epsilon \geq 0$ . We call  $\bar{f}^* \in \mathbb{R}$  an  $\epsilon$ -optimal solution to (7) if  $0 \leq \bar{f}^* - f^* \leq \epsilon$ .

As mentioned in the introduction, Alg. 1 relies on identifying non-decreasing, non-increasing, convex and concave segments of  $f(t)$  on  $[0, \Delta t]$  based on interval inclusions for the derivatives (10) and (11). As stated in the following proposition, such inclusions can be easily computed based on Thm. 2.

**Proposition 3.** Let  $[t] \subseteq [0, \Delta t]$  with  $|[t]| > 0$  and consider the interval matrix  $[C] = A[t]$ . Let  $k, l \in \mathbb{N}$  be such that  $2^l(k+2) > \| [C] \|_\infty$  and define  $[D]$  as in Thm. 2. Then, the interval inclusions

$$[f'] = [\underline{f}', \bar{f}'] = h^T [D] (Ax_0 + Bu_0), \quad \text{and} \quad (13)$$

$$[f''] = [\underline{f}'', \bar{f}''] = h^T A [D] (Ax_0 + Bu_0), \quad (14)$$

are such that

$$f'(t) \in [f'] \quad \text{and} \quad f''(t) \in [f''] \quad \text{for every } t \in [t]. \quad (15)$$

*Proof.* According to Thm. 2, we have  $\exp(At) \in [D]$  for every  $t \in [t]$ . Thus, (13) and (14) contain the r.h.s. in (10) and (11) for every  $t \in [t]$ , respectively. Consequently, (15) holds.  $\blacksquare$

Clearly, if  $\underline{f}' \geq 0$  results from (14), then  $f(t)$  is non-decreasing on the time-interval  $[t]$ . Analogously,  $\bar{f}' \leq 0$ ,  $\underline{f}'' \geq 0$ , or  $\bar{f}'' \leq 0$  guarantees  $f(t)$  to be non-increasing, convex, or concave on  $[t]$ , respectively. In each of these cases, it is easy to compute the local maximum of  $f(t)$  on  $[t]$ , i.e.,

$$f^\dagger := \max_{t \in [t]} f(t) \quad (16)$$

In fact,  $f(t)$  being convex, non-decreasing, or non-increasing implies  $f^\dagger = \max\{f(\underline{t}), f(\bar{t})\}$ ,  $f^\dagger = f(\underline{t})$ , or  $f^\dagger = f(\bar{t})$ . Finally, if  $f(t)$  is concave, solving (16) is a convex OP. In contrast, if  $\underline{f}' < 0 < \bar{f}'$  and  $\underline{f}'' < 0 < \bar{f}''$ , a straightforward computation of  $f^\dagger$  may not be possible. However, even in this case, the bounds on the derivatives can be used to compute an upper bound for the local maximum according to Def. 2 and Lem. 4.

**Definition 2.** Let  $[t] \subseteq [0, \Delta t]$  with  $w([t]) > 0$ . We call a function  $g : [t] \rightarrow \mathbb{R}$  a suitable overestimator for  $f$  on  $[t]$  if  $f(t) \leq g(t)$  for every  $t \in [t]$  and if the optimizer

$$t^\dagger := \arg \max_{t \in [t]} g(t) \quad (17)$$

can either be computed analytically or by solving a convex optimization problem.

**Lemma 4.** Let  $[t] \subseteq [0, \Delta t]$  with  $w([t]) > 0$  and assume  $[f']$  and  $[f'']$  with  $\underline{f}' < 0 < \bar{f}'$  and  $\bar{f}'' > 0$  are such that (15) holds. Then, the following three functions  $g : [t] \rightarrow \mathbb{R}$  are suitable overestimations for  $f$  on  $[t]$ .

1. The piecewise affine function

$$g(t) := \begin{cases} f(\underline{t}) + \overline{f}'(t - \underline{t}) & \text{if } t \leq t_c, \\ f(\overline{t}) - \underline{f}'(\overline{t} - t) & \text{otherwise,} \end{cases}$$

$$\text{where } t_c = \frac{\overline{f}'\underline{t} - \underline{f}'\overline{t} + f(\overline{t}) - f(\underline{t})}{\overline{f}' - \underline{f}'}$$

2. The piecewise quadratic function

$$g(t) := \begin{cases} f(\underline{t}) + f'(\underline{t})(t - \underline{t}) + \frac{\overline{f}''}{2}(t - \underline{t})^2 & \text{if } t \leq t_c, \\ f(\overline{t}) - f'(\overline{t})(\overline{t} - t) + \frac{\underline{f}''}{2}(\overline{t} - t)^2 & \text{otherwise,} \end{cases}$$

where

$$t_c := \begin{cases} \frac{0.5\overline{f}''(\overline{t}^2 - \underline{t}^2) + f'(\underline{t})\underline{t} - f'(\overline{t})\overline{t} + f(\overline{t}) - f(\underline{t})}{\overline{f}''(\overline{t} - \underline{t}) + f'(\underline{t}) - f'(\overline{t})} & \text{if } \frac{f'(\overline{t}) - f'(\underline{t})}{\overline{t} - \underline{t}} < \overline{f}'', \\ \overline{t} & \text{otherwise.} \end{cases}$$

3. The concave function  $g(t) := f(t) + \frac{\overline{f}''}{2}(t - \underline{t})(\overline{t} - t)$ .

The overestimators listed in Lem. 4 are adopted from [7], [8], and [9, Sect. 4]. In fact,  $\|f'\|$  and  $\|f''\|$  can be understood as local Lipschitz constants for  $f(t)$  and  $f'(t)$  as exploited in [7] and [8], respectively. We thus omit a detailed proof of Lem. 4 and refer to [7, 8, 9]. It is, however, important to note that the solution to (17) reads  $t^\dagger = t_c$  for the overestimator  $g$  of type 1. For type 2, we find  $t^\dagger \in \{\underline{t}, t_c, \overline{t}\}$ , which renders (17) trivial. Finally, for type 3, solving (17) is a convex OP. Based on Prop. 3 and Lem. 4, we are finally able to formulate an algorithm for the computation of an  $\epsilon$ -optimal solution to (7).

**Algorithm 1.** Solution of (7) via branch and bound.

1. Initialize the lower bound on the global maximum as  $\underline{f}^* \leftarrow h^T x_0$ . Initialize the list  $\mathcal{L}$  of tuples  $([t], [f^\dagger])$ , each containing a time-interval  $[t]$  and bounds  $[f^\dagger]$  on the local maximum of  $f$  on  $[t]$ , as  $\mathcal{L} \leftarrow \{([0, \Delta t], [-\infty, \infty])\}$ .
2. **for each** tuple  $([t], [f^\dagger])$  in  $\mathcal{L}$ , **for which the bounds on the local maximum read**  $[f^\dagger] = [-\infty, \infty]$ , **repeat the following steps.**
  - a) Compute  $[f']$  and  $[f'']$  according to Prop. 3 and define a suitable overestimator  $g$  for  $f$  on  $[t]$  (e.g., according to Lem. 4).
  - b) **if**  $\underline{f}' \geq 0$ , **set**  $[f^\dagger] \leftarrow [f(\overline{t}), f(\overline{t})]$ .  
**else if**  $\overline{f}' \leq 0$ , **set**  $[f^\dagger] \leftarrow [f(\underline{t}), f(\underline{t})]$ .  
**else if**  $\underline{f}'' \geq 0$ , **compute**  $f^\dagger = \max\{f(\underline{t}), f(\overline{t})\}$  **and set**  $[f^\dagger] \leftarrow [f^\dagger, f^\dagger]$ .  
**else if**  $\overline{f}'' \leq 0$ , **solve** (16) **and set**  $[f^\dagger] \leftarrow [f^\dagger, f^\dagger]$ .  
**else**, **solve** (17) **and set**  $[f^\dagger] \leftarrow [f(t^\dagger), g(t^\dagger)]$ .
  - c) **if**  $\underline{f}^\dagger > \underline{f}^*$ , **set**  $\underline{f}^* \leftarrow \underline{f}^\dagger$ .
3. Compute the upper bound  $\overline{f}^*$  on the global maximum by taking the maximum of all local upper bounds  $\overline{f}^\dagger$  of the tuples  $([t], [f^\dagger])$  in  $\mathcal{L}$ .

4. **if**  $w([f^*]) \leq \epsilon$ , **return**  $\bar{f}^*$  and terminate.
5. **for each** tuple in  $\mathcal{L}$  repeat the following step.
  - a) **if**  $\bar{f}^\dagger \leq \underline{f}^*$  and  $w([f^\dagger]) > \epsilon$ , remove tuple from  $\mathcal{L}$ .
6. Select the tuple  $([t], [f^\dagger])$  with the largest width  $w([f^\dagger])$  in  $\mathcal{L}$  and remove it from  $\mathcal{L}$ . Compute  $t_m = \frac{t + \bar{t}}{2}$  and insert the tuples  $([\underline{t}, t_m], [-\infty, \infty])$  and  $([t_m, \bar{t}], [-\infty, \infty])$  in  $\mathcal{L}$ . **go to** step 2.

In principle, Alg. 1 is similar to established branch and bound procedures for global optimization (see, e.g., [7], [10, Sects. 6 to 13], [8, Sect. 3], [9, Sect. 6], or [11, Sect. 3]). The main difference is that Alg. 1 makes use of bounds on the first *and* second derivative. First, this allows to identify a number of segments where the local maximum can be computed exactly. Second, it gives some flexibility w.r.t. the choice of suitable overestimators for the remaining segments. In fact, overestimators of type 1 (in Lem. 4) depend on  $[f']$  while type 2 and 3 build on  $[f'']$ . Regarding the computational effort, the strategy to compute both interval inclusions may be inefficient in general. Here, however, the simultaneous calculation of  $[f']$  and  $[f'']$  does not significantly increase the computational load compared to solely calculating  $[f']$  or  $[f'']$ . In fact, due to the special structure of  $f$ , we easily evaluate  $[f'] = h^T[d]$  and  $[f''] = h^T A[d]$  given the interval vector  $[d] := [D](Ax_0 + Bu_0)$ . Obviously, the computational effort to calculate  $[d]$  is dominated by the computation of the interval inclusion  $[D]$  for the matrix exponential.

As stated in Prop. 5, Alg. 1 is guaranteed to compute an  $\epsilon$ -optimal solution to (7) for every  $\epsilon > 0$ . In many cases, however, Alg. 1 is capable to solve (7) exactly, i.e., for  $\epsilon = 0$  (see Exmps. 1 through 3 in Sect. 4).

**Proposition 5.** *Let  $\epsilon > 0$  and let  $k, l \in \mathbb{N}$  be such that  $2^l(k+2) > \|A[0, \Delta t]\|_\infty$ . Then Alg. 1 terminates after finite time and returns an  $\epsilon$ -optimal solution to (7).*

*Proof.* It is easy to see that Alg. 1 provides an  $\epsilon$ -optimal whenever it terminates. Hence, it is sufficient to prove finite termination of the algorithm. Clearly, Alg. 1 terminates if (but not only if) we have  $w([f^\dagger]) \leq \epsilon$  for every tuple  $([t], [f^\dagger])$  in the list  $\mathcal{L}$ . In fact, the upper bound on the global maximum then satisfies

$$\bar{f}^* = \max_{([t], [f^\dagger]) \in \mathcal{L}} \bar{f}^\dagger \leq \max_{([t], [f^\dagger]) \in \mathcal{L}} \underline{f}^\dagger + \epsilon = \underline{f}^* + \epsilon,$$

i.e.,  $w([f^*]) \leq \epsilon$ . As a direct consequence, the time-interval  $[t]$  of a tuple  $([t], [f^\dagger])$  satisfying  $w([f^\dagger]) \leq \epsilon$  will never be bisected in step 6 of Alg. 1 (since this would contradict reaching step 6 after passing step 4 without termination). In the following, denote by  $[f'_0]$  and  $[f''_0]$  the interval inclusions for  $f'$  and  $f''$  on  $[0, \Delta t]$  and let  $j \in \mathbb{N}$  be such that

$$\max \left\{ w([f'_0]) \Delta \tau, \frac{w([f''_0])}{2} \Delta \tau^2, \frac{\bar{f}''_0}{8} \Delta \tau^2 \right\} \leq \epsilon, \quad (18)$$

where  $\Delta \tau := \frac{\Delta t}{2^j}$ . We obviously have

$$[0, \Delta t] = \bigcup_{i=0}^{2^j-1} [i, i+1] \Delta \tau \quad (19)$$

by construction. Consider any  $i \in \{0, \dots, 2^j - 1\}$ , set  $[t] = [i, i + 1] \Delta\tau$ , and note that  $w([t]) = \Delta\tau$ . Further note that the inclusions  $[f']$  and  $[f'']$  on  $[t]$  satisfy  $[f'] \subseteq [f'_0]$  and  $[f''] \subseteq [f''_0]$  since  $[t] \subseteq [0, \Delta t]$  (and since all involved operations are *inclusion increasing*; see [5] for details). Now assume an overestimator of type 1 (as in Lem. 4) is applied. We then find

$$\begin{aligned} g(t^\dagger) - f(t^\dagger) &\leq \max_{t \in [t]} g(t) - f(t) \\ &= \max_{t \in [t]} f(\underline{t}) + \overline{f}'(t - \underline{t}) - f(t) \\ &\leq \max_{t \in [t]} f(\underline{t}) + \overline{f}'(t - \underline{t}) - f(\underline{t}) - \underline{f}'(t - \underline{t}) \\ &= w([f']) w([t]) \leq w([f'_0]) \Delta\tau \leq \epsilon, \end{aligned}$$

where the first and second relation hold due to  $t^\dagger \in [t]$  and by definition of  $g$ , respectively. The third relation holds since

$$f(t) = f(\underline{t}) + \int_{\underline{t}}^t f'(\tau) d\tau \geq f(\underline{t}) + \underline{f}'(t - \underline{t})$$

for every  $t \in [t]$ . Finally, the last relations hold due to  $[f'] \subseteq [f'_0]$  and according to (18). Using analogous arguments, we obtain  $g(t^\dagger) - f(t^\dagger) \leq \epsilon$  also for overestimators of type 2 or 3. We thus find  $w([f^\dagger]) \leq \epsilon$  for the bounds on the local maximum of  $f$  on  $[t]$  according to step 2.(b) of Alg. 1. Since  $i \in \{0, \dots, 2^j - 1\}$  was arbitrary, this observation holds for every time interval  $[i, i + 1] \Delta\tau$  on the r.h.s. of (19). As a consequence, the number of required bisections in step 6 of Alg. 1 is limited and the algorithm terminates after finite time. To see this, first note that  $j$  and  $i$  can be understood as the height and the position of a leaf node in a perfect binary tree, respectively. The binary tree can be associated with the bisection procedure. In fact, every inner node can be linked to the bisection of a time-interval. Now, the perfect binary tree with height  $j$  refers to the worst-case scenario, where the bisection continues until we obtain the partition on the r.h.s. of (19). Since this tree contains  $\sum_{i=0}^{j-1} 2^i = 2^j - 1$  inner nodes, we obtain a maximum of  $2^j - 1$  bisections.  $\blacksquare$

## 4 Numerical Examples

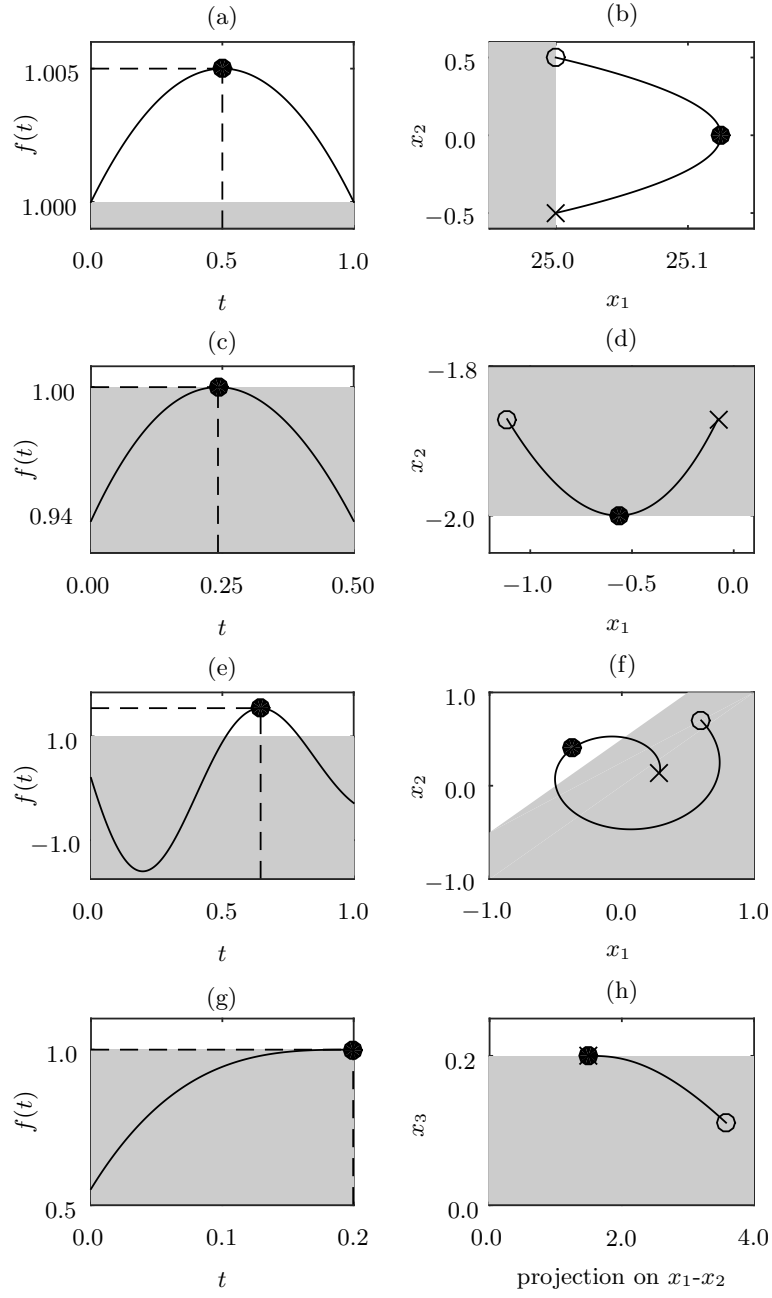
We analyze four examples in the following. The first two examples address technical systems taken from [12] and [1]. In contrast, Exmp. 3 and 4 are of academic nature. In fact, these examples were purely designed to challenge Alg. 1.

The application of Alg. 1 requires to specify an error bound  $\epsilon$ . Moreover, the underlying computation of interval inclusions for matrix exponentials depends on the parameters  $k, l \in \mathbb{N}$  (see Thm. 2). We set  $\epsilon = 10^{-6}$  and  $k = l = 10$  for all examples.

**Example 1.** *We first analyze the double integrator in [12] with the system matrices*

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$





**Figure 1:** Illustration of  $f(t)$  (left figures) and  $\varphi(t)$  (right figures) for Exmps. 1 through 4 (from top to bottom). In each figure, the point where the maximum  $f^*$  is attained is marked with a filled circle. Open circles and crosses refer to initial states  $x_0$  and final states  $\varphi(\Delta t)$ , respectively. State constraints are violated outside the gray regions.

and the constraints  $\mathcal{X} = \{x \in \mathbb{R}^2 \mid |x_1| \leq 25, |x_2| \leq 5\}$  and  $\mathcal{U} = [-1, 1]$ . As in [12], we consider the sampling time  $\Delta t = 1$  and obtain the discretized system matrices

$$\hat{A} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad \hat{B} = \begin{pmatrix} 0.5 \\ 1.0 \end{pmatrix}.$$

Obviously, for the initial state  $x_0 = (25.0 \ 0.5)^T \in \mathcal{X}$ , the only input  $u_0 \in \mathcal{U}$  for which the discretized system satisfies the state constraints at the next sampling instant, i.e., for which  $\hat{A}x_0 + \hat{B}u_0 \in \mathcal{X}$ , is  $u_0 = -1$ . In fact, for any  $u_0 \in (-1, 1]$ , the state constraint  $x_1 \leq 25$  will be violated. However, even for the choice  $u_0 = -1$ , the continuous-time system may violate the state constraints for some  $t \in (0, \Delta t)$ . To check whether the constraint  $x_1 \leq 25$  will be violated (or not), we set  $h = (0.04 \ 0.00)^T$  and solve (7). Clearly, since  $A$  is nilpotent, (7) can be easily solved analytically. We initially ignore this observation and apply Alg. 1.

Following the steps in Alg. 1, we first initialize the lower bound for the global maximum as  $\underline{f}^* = h^T x_0 = 1$  and the list of tuples as  $\mathcal{L} = \{([0, \Delta t], [-\infty, \infty])\}$ . Since  $[f^\dagger] = [-\infty, \infty]$ , we then evaluate inclusions for  $f'$  and  $f''$  on  $[0, \Delta t]$  in step 2.(a) and obtain the (exact) intervals

$$[f'] = [-0.02, 0.02] \quad \text{and} \quad [f''] = [-0.04, -0.04].$$

Since  $\bar{f}'' \leq 0$ , the algorithm recognizes that  $f$  is concave in step 2.(b), solves the convex OP  $f^\dagger = \max_{t \in [0, \Delta t]} f(t) = 1.005$ , and sets  $[f^\dagger] = [f^\dagger, f^\dagger]$ . Now, due to  $\underline{f}^\dagger = 1.005 > \underline{f}^*$ , the lower bound on the global maximum is updated in step 2.(c). Since  $([0, \Delta t], [1.005, 1.005])$  is the only tuple in  $\mathcal{L}$ , we move to step 3 and set  $\bar{f}^* = 1.005$ . Finally, the algorithm terminates in step 4 since  $w([f^*]) = 0 \leq \epsilon$ .

For this example, it is easy to verify the computed result by analytically solving (7). In fact, since  $A$  is nilpotent with degree  $r = n = 2$ , we obtain

$$\begin{aligned} f(t) &= h^T x_0 + h^T (Ax_0 + Bu_0)t + 0.5 h^T ABu_0 t^2 \\ &= 1 + 0.02t - 0.02t^2 \end{aligned}$$

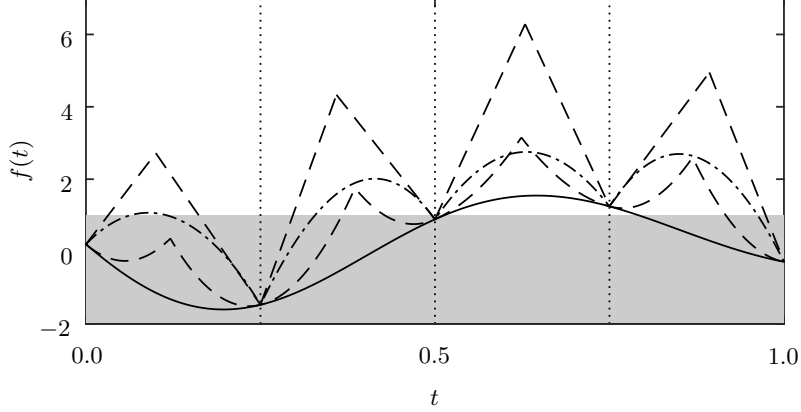
according to (12). We thus find  $f^* = f(0.5) = 1.005 = \bar{f}^* = \underline{f}^*$ . Clearly, since  $f^* > 1$ , the continuous-time system will violate the state constraint for some (here all)  $t \in (0, \Delta)$ . This can also be observed in Figs. 1.(a) and 1.(b), where  $f(t)$  and  $\varphi(t)$  are illustrated, respectively.

**Example 2.** We consider the example in [1] with

$$A = \begin{pmatrix} -0.7 & 0.1 \\ 2.0 & -0.1 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 2.0 \\ 1.0 \end{pmatrix}$$

plus  $\mathcal{X} = \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq 2\}$  and  $\mathcal{U} = [-1, 1]$ . As in [1], the sampling time is chosen as  $\Delta t = 0.5$ . We analyze whether the continuous-time system violates the constraint  $x_2 \geq -2$  for the initial state  $x_0 = (-1.1135 \ -1.8708)^T$  and the input  $u_0 = 0.9355$ .

To this end, we solve (7) with  $h = (0.0 \quad -0.5)^T$  and obtain  $\bar{f}^* = \underline{f}^* = 0.9999$  using Alg. 1. Thus, the continuous-time system does not violate the state constraint  $x_2 \geq -2$  for any  $t \in [0, \Delta]$ . This observation is important, since  $(x_0 \ u_0)^T$  marks a vertex of the adapted constraint set  $\hat{\mathcal{Z}}$  as computed in [3, Sect. IV]. In other words,  $f^* \leq 1$  is required to confirm the results in [3]. An illustration of  $f(t)$  and  $\varphi(t)$  can be found in Figs. 1.(c) and 1.(d), respectively.



**Figure 2:** Illustration of some overestimators for  $f(t)$  as in Exmp. 3 after three bisections in Alg. 1. The dashed lines refer to the piecewise linear and quadratic overestimators as introduced in Lem. 4 (type 1 and 2), respectively. The dash-dotted curves show the concave overestimators (type 3).

**Example 3.** We consider the system matrices

$$A = \begin{pmatrix} -1 & 7 \\ -7 & -1 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} -1 \\ 0 \end{pmatrix},$$

the constraints  $\mathcal{X} = \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq 1, -2x_1 + 2x_2 \leq 1\}$  and  $\mathcal{U} = [-1, 1]$ , and the sampling time  $\Delta t = 1.0$ . To check whether the continuous-time systems violates the constraint  $-2x_1 + 2x_2 \leq 1$  for  $x_0 = (0.6 \ 0.7)^T$  and  $u_0 = 1.0$ , we solve (7) with  $h = (-2 \ 2)^T$  and obtain  $\bar{f}^* = \underline{f}^* = 1.5465$  using Alg. 1. Thus, the continuous-time systems violates the state constraints for some  $t \in (0, \Delta t)$  as confirmed in Figs. 1.(e) and 1.(f). In contrast to Exmps. 1 and 2, Alg. 1 does not terminate without any bisection. In fact, as itemized in Tab. 1, we require eight bisections and the solution of three convex OP to identify  $\bar{f}^*$  using the second overestimator proposed in Lem. 4. A snapshot of the computed overestimators after three bisections is shown in Fig. 2.

**Example 4.** We consider the system matrices

$$A = \begin{pmatrix} 0 & 6 & 5 \\ 5 & 1 & 0 \\ 3 & 2 & 1 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 1 \\ 0 \\ -2 \end{pmatrix},$$

the constraints  $\mathcal{X} = \{x \in \mathbb{R}^n \mid \|x\|_\infty \leq 0.2\}$  and  $\mathcal{U} = [-1, 1]$ , and the sampling time  $\Delta t = 0.2$ . To check whether the continuous-time systems violates the constraint  $x_3 \leq 0.2$  for  $x_0 = (2.6724 \quad -2.3762 \quad 0.1105)^T$  and  $u_0 = 1.0$ , we solve (7) with  $h = (0 \ 0 \ 5)^T$  and obtain  $\bar{f}^* = 1.0000$  using Alg. 1. In contrast to Exmps. 1 through 3, the result  $\bar{f}^* = 1.0000$  is not guaranteed to be exact. In fact, we obtain  $\bar{f}^* - \underline{f}^* = 0.3123 \cdot 10^{-6}$  using the second overestimator in Lem. 4. The inexactness can be explained as follows. The example is constructed in such a way that  $f^* = f(\Delta t)$  and  $f'(\Delta t) = f''(\Delta t) = 0$ . In other words, the maximum on  $[0, \Delta t]$  is a saddle point of  $f(t)$ . Thus, for any time-interval containing  $\Delta t$ , one of the interval inclusions  $[f']$  and  $[f'']$  has to be exact (at least  $\underline{f}'$  or  $\bar{f}''$ ) in order to identify  $f$  being non-decreasing or concave. However, since interval inclusions are inexact in general and in particular for this example,  $\bar{f}^*$  has to be identified solely by using the overestimators  $g$ . Consequently, the number of required bisections is high compared to Exmps. 1 through 3 (see Tab. 1).

**Table 1:** Statistics on the application of Alg. 1 to Exmps. 1 through 4. For every example and every overestimator  $g$  as in Lem. 4, we list the number of bisections and the number of solved convex OP necessary to identify  $\bar{f}^*$ . The itemized errors refer to  $(\bar{f}^* - \underline{f}^*) \cdot 10^6$ .

	$g$	bisections	convex OP	error
Exmp. 1	1–3	0	1	0
Exmp. 2	1–3	0	1	0
Exmp. 3	1	11	4	0
	2	8	3	0
	3	7	15	0
Exmp. 4	1	109	90	0.8149
	2	15	0	0.3123
	3	14	29	0.3022

## 5 Conclusion

We presented a numerical method for the rigorous verification of constraint satisfaction for sampled linear systems. In particular, we proposed a tailored branch and bound algorithm for the solution of the non-convex OP (7) (resp. (5)). The core of the algorithm is a recently published procedure for the inclusion of interval matrix exponentials (see [5]). Being able to solve (5) for different  $x_0$  and  $u_0$  allows us to (offline) compute adapted state and input constraints according to [2, Prop. 4 and Thm. 5] or [3, Prop. 2]. Satisfying these adapted constraints for the discretized system (4) finally

guarantees constraint satisfaction of the continuous-time system (1) w.r.t. the original constraints (2).

The new method was illustrated with four examples. For every example, we were able to compute an  $\epsilon$ -optimal solution to the non-convex OP (7) (with  $\epsilon = 10^{-6}$ ). For three examples, the OP has even been solved exactly. For the two technical examples taken from [12] and [1], the algorithm terminated instantaneously without branching (i.e., without bisections). In fact, branching (and bounding) was only required for the two academic examples, which were designed to challenge Alg. 1. Such challenges are unlikely to appear in practice, however, since they were either caused by an inappropriately high sampling time  $\Delta t$  (see Fig. 1.(f)) or an extremely rare feature of  $f$  in terms of a saddle point at the boundary of  $[0, \Delta t]$  (see Fig. 1.(g)).

Algorithm 1 was particularly designed to solve problems of the form (7). However, it can be used to solve any univariate OP on a convex domain, for which the objective function  $f$  is of class  $\mathcal{C}^2$  and for which interval inclusions for the first *and* second derivative of  $f$  can be computed efficiently. In this context, note that the list of suitable overestimators in Lem. 4 is (by far) not complete. The overestimator of type 2, which performed most successfully for the analyzed examples (see Tab. 1 and Fig. 2) can for example be further improved using the results in [11].

## 6 Acknowledgments

Financial support by the German Research Foundation (DFG) through the grant SCHU 2094/1-1 is gratefully acknowledged.

## References

## References

- [1] P. Sopasakis, P. Patrinos, H. Sarimveis, MPC for sampled-data linear systems: Guaranteeing constraint satisfaction in continuous-time, *IEEE Trans. Autom. Control* 59 (4) (2014) 1088–1093.
- [2] L. Berardi, E. De Santis, M. D. Di Benedetto, G. Pola, Controlled safe sets for continuous time systems, in: *Proc. of European Control Conference*, 2001, pp. 803–808.
- [3] M. Schulze Darup, Efficient constraint adaptation for sampled linear systems, in: *Proc. of the 54th Conference on Decision and Control*, 2015.
- [4] R. E. Moore, *Methods and applications of interval analysis*, SIAM, 1979.
- [5] A. Goldsztejn, A. Neumaier, On the exponentiation of interval matrices, *Reliable Computing* 20 (2014) 52–72.

- [6] O. Kosheleva, V. Kreinovich, G. Mayer, H. T. Nguyen, Computing the cube of an interval matrix is NP-hard., in: Proc. of the 2005 ACM Symposium on Applied Computing, 2005, pp. 1449–1453.
- [7] S. A. Piyavskii, An algorithm for finding the absolute extremum of a function, Com. Maths. Math. Phys. 12 (1972) 57–67.
- [8] L. Breiman, A. Cutler, A deterministic algorithm for global optimization, Mathematical Programming 58 (1993) 179–199.
- [9] C. D. Maranas, C. A. Floudas, Global minimum potential energy conformations of small molecules, Journal of Global Optimization 4 (2) (1994) 135–170.
- [10] E. R. Hansen, Global optimization using interval analysis: The one-dimensional case, J. Optim. Theory Appl. 29 (3) (1979) 251–293.
- [11] Y. D. Sergeyev, Global one-dimensional optimization using smooth auxiliary functions, Mathematical Programming 81 (1) (1998) 127–146.
- [12] P. O. Gutman, M. Cwikel, An algorithm to find maximal state constraint sets for discrete-time linear dynamical systems with bounded control and states, IEEE Trans. Autom. Control 32 (3) (1987) 251–253.